sensors

Article

A BERT Framework to Sentiment Analysis of Tweets

Abayomi Bello, Sin-Chun Ng and Man-Fai Leung *

School of Computing and Information Science, Faculty of Science and Engineering, Anglia Ruskin University, Cambridge CB1 1PT, UK
* Correspondence: man-fai.leung@aru.ac.uk

Abstract: Sentiment analysis has been widely used in microblogging sites such as Twitter in recent decades, where millions of users express their opinions and thoughts because of its short and simple manner of expression. Several studies reveal the state of sentiment which does not express sentiment based on the user context because of different lengths and ambiguous emotional information. Hence, this study proposes text classification with the use of bidirectional encoder representations from transformers (BERT) for natural language processing with other variants. The experimental findings demonstrate that the combination of BERT with CNN, BERT with RNN, and BERT with BiLSTM performs well in terms of accuracy rate, precision rate, recall rate, and F1-score compared to when it was used with Word2vec and when it was used with no variant.

Keywords: sentiment analysis; deep learning; tweets; BERT; LSTM; CNN

Citation: Bello, A.; Ng, S.-C.; Leung, M.-F. A BERT Framework to Sentiment Analysis of Tweets. *Sensors* **2023**, *23*, 506. https://doi.org/10.3390/s23010506

Academic Editor: Carmela Comito

Received: 17 November 2022
Revised: 26 December 2022
Accepted: 28 December 2022
Published: 2 January 2023

1. Introduction

It is obvious that the emergence of real-time information networking platforms such as Twitter has led to the development of an unmatched public collection of viewpoints about all relevant worldwide entities thereby interfering and affecting human lifestyle [1]. Twitter may be a great platform for opinion generation and presentation, but it also presents new and unique obstacles, and the process would be incomplete without capable tools for assessing those thoughts to speed up their consumption. The best approach over time has proven to be using sentiment analysis tools to identify individual attitudes and emotions [2].

Sentiment analysis, which is additionally referred to as subjective investigation or artificial intelligence of emotions) is a natural language processing (NLP) technique for extracting information patterns and key characteristics from a large body of text. It analyses the thoughts, attitudes, viewpoints, opinions, convictions, remarks, requests, inquiries, and interests expressed by the author based on feelings not reason in the form of texts, with entities such as service, issue, person, product, event, object, organizations, and their attributes. It identifies the author's overall attitudes toward a text, which could be anything from blog posts to product reviews to online forums to speeches to data from databases to social media documents [3].

The need for natural language processing (NLP) arises as a result of the need for computers to understand the spoken and written language of humans. This brought about bag of words (BOW) which uses N-grams but the contextual meaning of words is ignored with BoW models, after which the word embedding was developed to overcome this issue which always takes words similar in meaning as similar contexts but this model always relies on large vocabulary and high computational power [4]. Word2Vec was then developed which generates only one vector embedding for each word but also has a shortcoming by considering left or right context. The transformer model which was introduced by Google in 2018 then solves the aforementioned problem and enhances language processing. It helped to overcome the problem of transfer learning and has recorded great achievement on natural language processing tasks such as named entity recognition (NER) and question answering and sentiment analysis. The BERT has been pre-trained on large corpus of English data which acts like a benchmark and helps solve similar problems. The BERT has transformer encoder layers enhanced with a self-attention mechanism.

This study seeks to produce an approach that can identify the opinion and attitude of a writer in a tweet according to context. The pre-trained transformer BERT and Word2Vec was used with the convolutional neural network (CNN), recurrent neural network (RNN), bidirectional long short-term memory (BiLSTM), and the experiment was carried out and the Proposed BERT obtained a state-of-the-art performance.

This study's contributions include proposing the BERT for NLP that can identify sentiment in tweets according to three categories (positive, negative, and neutral) based on the context of the writer. Our approach is distinct from similar studies because:

- We have trained our model using six different datasets which is a combination of different tweets.
- We combined knowledge embedded in the pre-trained bidirectional transformer (BERT) with a deep learning classifier to detect sentiment (positive, negative, or neutral) other than just using a machine learning classifier.
- The proposed BERT will dynamically generate a vector according to the word context and when placed into deep learning classifiers such as CNN, RNN, or BiLSTM to predict output, achieves an accuracy of 93% and F-measure of 95%.

2. Literature Review

The development of the internet has changed how people now express their ideas and thoughts. According to Kepios, there are 4.74 billion social media users around the world equating to 59.3 percent of the total global population. For context, the data suggest that more than 75 percent of the eligible global population now uses social media [5]. Further research shows that a typical social media user actively uses or visits an average of 7.2 million different social platforms each month and spends close to 2 and half hours per day using social media. The world spends more than 10 billion hours using social platforms each day which is equivalent to nearly 1.2 million years of human existence. Additionally, social media gives businesses a chance by offering them a platform to engage with their customers for advertising. People heavily rely on user-generated content from the internet when making decisions. Social networking services such as Twitter are a valuable source of information to find out what happened or what is happening in a geographic area [6]. Microblogs have become an important origin of information regarding events happening in a location during a period of time [7]. Twitter is one of the most used platforms with easy access to tweets with connection to the API and having a maximum length of 280 characters making it suitable to effectively monitor emotions, sentiments, opinions, and attitudes of different subject matter.

Artificial Intelligence (AI) is the art and science of building intelligent machines, particularly smart computer programs. Furthermore, AI can be defined as the imitation or reproduction of cognitive functions by computer systems that can reason logically and act in ways resembling those of humans. The subject gained popularity as a subject in academic literature after the 1950s. Various industries use AI, including communication, IT, healthcare, agriculture, logistics, education, and aviation.

In recent years, natural language processing (NLP) has drawn a lot of interest for its ability to computationally represent and analyze human language. It has expanded the range of industries in which it is used, including machine translation, email spam detection, information extraction, summarization, and medicine [8]. It makes interactions between people and computers simple and effective using computational linguistics and machine learning. NLP systems can output written texts or processed speech from inputs such as text, images, or speech [9].

Neural network is a subset of machine learning with numerous applications such as compressed image reconstruction [10], asset allocation [11,12], non-negative matrix factorization [13,14], and model predictive control [15]. With the development of transfer learning, G.E. Hinton introduced the concept of deep learning, and it is simply extracting features from raw data with the help of using layers [16]. The human brain affects neural networks, which are made up of numerous neurons and form amazing networks. Deep learning networks can be used to teach both supervised and unsupervised categories [17–20]. CNN, RNN, and many other networks with more than three layers are considered deep learning approaches. Text creation, vector representation, word representation estimation, sentence classification, phrase modeling, feature presentation, and emotion recognition benefit greatly from neural networks [21,22].

Additionally, the term deep learning has gained popularity among computer scientists to refer to pattern-recognition algorithms that enable computers to learn on their own, leading to improvements in speech and image recognition as well as more precise translation software. In addition to a deeper focus on context, thought, and abstraction, deep learning can also refer to knowledge that is less surface level and more contemplative and abstract [23].

2.1. Sentiment Analysis Based on Machine Learning Approach

Suhasini et al. [4] were able to identify emotions on Twitter using supervised learning. K-nearest neighbor (KNN) and naive Bayes (NB) were the two algorithms compared and the study shows that naive Bayes outperformed the K-nearest neighbor.

Jayakody et al. [1] collected data from twitter posts based on product review, then analyzed using the support vector machine (SVM), logical regression, and K-nearest neighbor machine learning algorithm and count vectorizer and term frequency-inverse document frequency mechanisms for converting text into vectors for the data to be inputted into the machine learning model. The highest accuracy score was achieved by logistic regression with a count vectorizer with an accuracy rate of 88.26%.

Bhagat et al. [24] used a hybrid approach of naive Bayes and K-nearest neighbor to divide tweets into three classes: positive, negative, and neutral, and they achieved a better accuracy than the random forest.

2.2. Sentiment Analysis Based on Deep Learning and BERT Approach

Chiorrini et al. [25] proposed two BERT-based approaches for text classification: BERT-base and cased BERT-base. They gathered information from microblogging sites, particularly twitter. Two separate datasets were employed in their experiment, and they were used for sentiment analysis and emotion recognition. The proposed model gives an accuracy of 92%. They emphasized that BERT produces positive outcomes for text classification.

Huang et al. [26] presented a model for text classification where he used the deep convolutional neural network, bidirectional gated recurrent. This model was based on BERT. Two datasets (CCERT email and movie comment) were used. The result gave an accuracy of 92.66% on CCERT and 91.89% on the movie dataset.

The researchers of [27] represented a seven-layer framework to analyze the feelings of sentences. This framework was based on CNN and Word2vec to calculate vector representation and SA, respectively. Google has proposed the use of Word2vec. To improve the correctness and generalizability of the suggested model, the researcher employed the dropout technology, normalization, and parametric rectified linear unit (PReLU). The framework was validated using a data set from rottentomatoes.com that contains a corpus of movie review extracts with five labels: positive, slightly positive, neural, negative, and somewhat negative. Compared to earlier models, such as matrix-vector recursive neural network (MV-RNN) and recursive neural network, the suggested model outperformed the previous with an accuracy of 45.4%.

3. Materials and Methods

This section provides a concise and precise description of the experimental results, their interpretation, as well as the experimental conclusions that can be drawn.

3.1. Data Collection and Preprocessing

Six datasets were used and they were tweets collected from Kaggle. The six datasets were combined using Python's concatenating function. There are 2 columns and 212,661 rows in the dataset altogether. The null values were eliminated, and mapping was completed. 1.0 was assigned to positive, 0.0 to neutral, and −1.0 to negative. Special characters, punctuation, numbers, symbols, and hashtags were removed from the model dataset as shown in Table 1 below. The dataset sentiment contains 71,658 neutrals, 85,231 positives, and 55,772 negatives and has a percentage of 40.1% to be positive, 33.7% neutral, and 26.2% negative as shown in Table 2 below. The raw tweets were preprocessed and fed into different models as shown in Figures 1–3.

Table 1. Dataset composition.

Dataset Name	Rows by Columns	Source
Twitter_Data.csv	162,980 × 2	Kaggle.com
Apple-twitter-sentiment-texts.csv	1630 × 2	Kaggle.com
FinalSentimentdata2.csv	3090 × 2	Kaggle.com
Tweets.csv	14,640 × 2	Kaggle.com
Train.csv	27,481 × 4	Kaggle.com
Test.csv	3534 × 3	Kaggle.com
Model Dataset after Concatenating	213,355 × 2	

Table 2. Sentiment count.

Sentiment	Sentiment Count	Sentiment Percentage
Negative	55,717	26.2%
Neutral	71,658	33.7%
Positive	85,167	40.1%

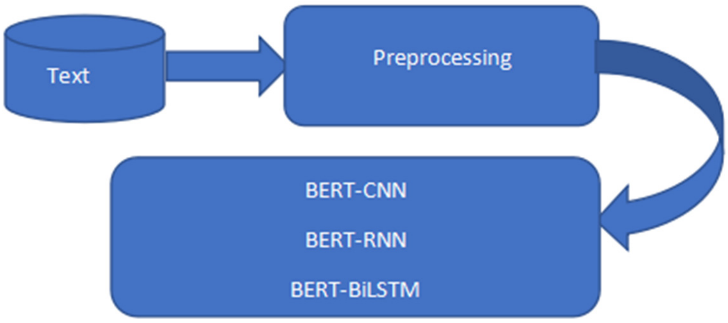

Figure 1. BERT-based architecture.

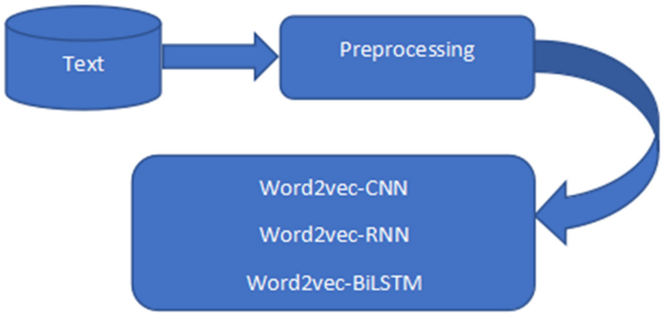

Figure 2. Word2Vec-based architecture.

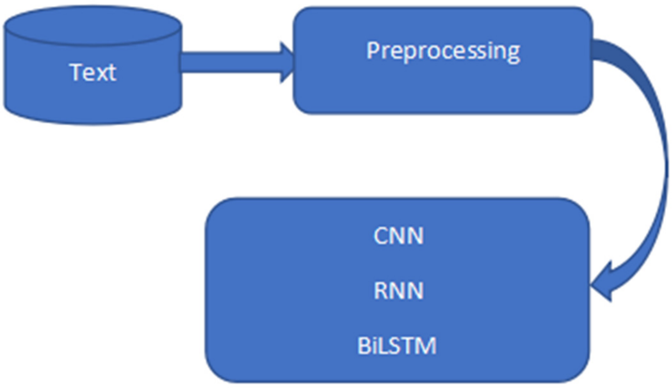

Figure 3. CNN, RNN, BiLSTM architecture.

3.2. Bidirectional Encoder Representations from Transformers (BERT)

BERT was developed in 2018 for natural language understanding tasks to assist machines in comprehending the context of phrases [28]. It employs transfer learning, and the architecture is based on the transformer model. Transfer learning entails training a model for a broad task and then applying the knowledge gained to fine-tune BERT for a new task [29]. BERT has been trained on two tasks: masked language tasks, in which sentences are fed to the model and some words are masked or hidden for the model, and the model attempts to predict these hidden words, and unmasked language tasks, in which sentences are fed to the model and some words are masked or hidden for the model, and unmasked language tasks, in the other task is sentence prediction where pair of sentences are fed to the model each round and the model need to predict whether one sentence is followed by the other or not. BERT has been trained over a large dataset for these two tasks. The dataset contained all English Wikipedia and 11,038 books. BERT uses an encoder from a transformer model, a type of neural network that takes a sentence as input to the model. Then each word of the sentence is tokenized, and these tokenized words are fed to the BERT model. BERT output is a vector representation for each tokenized word.

Using encoders from Transformer enables BERT to have a better context understanding than traditional neural networks such as LSTM or RNN since the encoder process all inputs, which is the whole sentence, simultaneously so when building a context for a word, BERT will take into account the inputs before it and also the inputs after the word, while the LSTM or RNN process the input taking in account only the prior inputs, and that will be reflected on the output vector value for the word, so the word "python" in the two sentences (I just started learning Python) and (Python are found in part of Africa and Asia) would have the same vector value—and as a result the same meaning—when using LSTM or RNN; on the other hand, it would have two different vectors using BERT, so as a result, using BERT will in most cases give us better performance than using the traditional machine learning algorithms.

3.3. Word Embedding

Computers are programmed to operate in numbers. It worked on Bits which are zeros and ones. The question now is what happens when the software or a task must process a word? This word needs to be given to the computer as the only thing it can understand is numbers which means it needs to be broken down into bits (zeros and ones). The most straightforward approach in NLP is to create a vocabulary with many terms and assign a number to each word in the vocabulary [30]. Word embedding analysis is a natural language processing method in which a neural network is trained using machine learning to anticipate the contexts in which words are employed [28].

3.4. RNN

RNN is a type of artificial neural network that identifies patterns in data and utilizes them to anticipate the following most likely outcome. It operates on the tenet that each layer's output is saved and fed back into the system's input in order to forecast that layer's output.

For the RNN, the learning rate was set to be 0.01, on 10 epochs using Adam optimizer with batch size of 128, activation was softmax and categorical crossentropy was used as the loss as shown in Table 3 below.

Table 3. Parameter setting of RNN.

Parameter	Values
Learning rate	0.001
Epoch	10
Optimizer	Adam
Batch size	128
Activation	Softmax
Loss	categorical_crossentropy

3.5. CNN

CNN is formed of different layers of neurons. This works best on images, when an image is entered, each layer of the network creates a number of activations that are passed on to the following layer. Typically, the first layer extracts fundamental features such as edges that run horizontally or diagonally. The following layer receives this output and detects more intricate features such as corners or multiple edges. The network may recognize increasingly more complex elements, including objects, faces, etc., as we go further into it.

For the CNN, the model was trained on 10 epochs using Adam optimizer with batch size of 128, the activation was softmax and categorical crossentropy was used as the loss as shown in Table 4 below.

Table 4. Parameter setting of CNN.

Parameter	Values
Epoch	10
Optimizer	Adam
Batch size	128
Activation	Softmax
Loss	categorical_crossentropy

3.6. BiLSTM

A bidirectional long short-term memory, or BiLSTM, was employed since it is a model in which processing is done in order. It comprises dual LSTMs, one is open to input in one direction, while the other is open to input in another. A fully connected neural network is made up of multiple fully connected layers that link all of the neurons in each layer to all of the neurons in the other layer [31].

For the BiLSTM, the model was trained on 10 epochs using Adam optimizer with batch size of 128, activation was softmax and categorical crossentropy was used as the loss as shown in Table 5 below.

Table 5. Parameter setting of BiLSTM.

Parameter	Values
Epoch	10
Optimizer	Adam
Batch size	128
Activation	Softmax
Loss	categorical_crossentropy

3.7. Word2Vec

For each word, the Word2Vec embedding approach only offers a single, independent embedding vector. Only one vector per word is saved by Word2vec in the output model. Although Word2vec is trained using contextual neighbors, a downstream NLP task uses it without context, since the representation is only kept as one vector per word. Therefore, stagnant in use. This restricts the ability to understand a word's meaning across two contexts and used in two different situations, for example, "river bank" and "bank deposit", "apple macbook" and "apple as a fruit" or "Python" as a programming language and "python" as a snake.

For the Word2vec with CNN, RNN and BiLSTM, the model was trained on 10 epochs with learning rate 1×10^{-5} using Adam optimizer with batch size of 32, activation was softmax and categorical crossentropy was used as the loss as shown in Table 6 below.

Table 6. Parameter setting of Word2vec with CNN, RNN, and BiLSTM.

Parameter	Values
Learning rate	1×10^{-5}
Epoch	10
Optimizer	Adam
Batch size	32
Activation	Softmax
Loss	categorical_crossentropy

3.8. BERT

BERT has been trained on two tasks: masked language tasks, in which sentences are fed to the model and some words are masked or hidden for the model, and the model attempts to predict these hidden words, and unmasked language tasks, in which sentences are fed to the model and some words are masked or hidden for the model, and unmasked language tasks, in the other task is sentence prediction where pair of sentences are fed to the model each round and the model need to predict whether one sentence is followed by the other or not. BERT has been trained over a large dataset for these two tasks. The dataset contained all English Wikipedia and 11,038 books. BERT uses an encoder from a transformer model, a type of neural network that takes a sentence as input to the model. Then each word of the sentence is tokenized, and these tokenized words are fed to the BERT model. BERT output is a vector representation for each tokenized word.

Using encoders from Transformer enables BERT to have a better context understanding than traditional neural networks such as LSTM or RNN since the encoder process all inputs, which is the whole sentence, simultaneously so when building a context for a word, BERT will take into account the inputs before it and also the inputs after the word, while the LSTM or RNN process the input taking in account only the prior inputs, and that will be reflected on the output vector value for the word, so the word "python" in the two sentences (I just started learning Python) and (Python are found in part of Africa and Asia) would have the same vector value—and as a result the same meaning—when using LSTM or RNN; on the other hand, it would have two different vectors using BERT, so as a result, using BERT will in most cases give us better performance than using the traditional machine learning algorithms. The BERT comes in two forms which are the BERT base and the BERT large.

The BERT base consists of 12 encoders with a hidden size of 768 and the BERT large has 24 encoders with a hidden size of 1024. The study employed the BERT base.

For the BERT with CNN, RNN and BiLSTM, the model was trained on 10 epochs with learning rate 1×10^{-5} using Adam optimizer with batch size of 128, the activation was softmax and categorical crossentropy was used as the loss as shown in Table 7 below.

Table 7. Parameter setting for BERT with CNN, RNN, and BiLSTM.

Parameter	**Values**
Learning rate	1×10^{-5}
Epoch	10
Optimizer	Adam
Batch size	128
Activation	Softmax
Loss	sparse_categorical_crossentropy

4. Results and Discussion

4.1. Performance Indicators

Precision is a measure of correctness that explains how many total positive predictions are positive. It is determined by dividing the total number of predicted positives by the total number of classified positives. The precision level should be high for a well-performed model. Precision is defined as follows:

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}), \tag{1}$$

where TP is true positive and FP is false positive.

A recall is the ratio of all positively classified classes that were correctly identified to all positively classified classes or the number of classes with a positive outcome that are correctly predicted. A good model should have a high recall rate. Recall is defined as follows:

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}), \tag{2}$$

where FN is false negative.

A high F1-score indicates high precision and recall because the score contains information about these two variables. It is defined as follows:

$$\text{F1} = (2 \times \text{Precision} \times \text{Recall})/(\text{Precision} + \text{Recall}). \tag{3}$$

Mean absolute error describes the discrepancy between actual and anticipated values. As the value goes down, the model's performance gets better. An ideal predictor of the outputs is a model with a mean absolute error of zero.

The average of the square of the difference between the data's original and anticipated values is used to calculate the mean square error. As the value goes down, the model's performance gets better.

The standard deviation of the errors that result from making a prediction on a dataset is known as the root mean square error. However, when determining the model's accuracy, the value's root is considered. As the value goes down, the model's performance gets better.

4.2. Experimental Results

Table 8 shows the summary of the accuracy, precision, recall, and F1-score of all the models considered in the study.

Table 9, below, shows the mean absolute error, mean squared error, and root mean square error of all the models considered in the study.

Table 8. Model performance summary.

Models	Accuracy	P	P	P	R	R	R	F1	F1	F1
		NEG	NEU	POS	NEG	NEU	POS	NEG	NEU	POS
CNN	89%	0.88	0.88	0.90	0.81	0.90	0.93	0.84	0.89	0.91
RNN	90%	0.89	0.89	0.92	0.86	0.90	0.94	0.87	0.90	0.93
BiLSTM	90%	0.87	0.89	0.93	0.88	0.90	0.92	0.88	0.89	0.93
Word2Vec-CNN	57%	0.50	0.60	0.60	0.45	0.53	0.70	0.47	0.56	0.64
Word2Vec-RNN	48%	0.50	0.47	0.48	0.30	0.40	0.67	0.37	0.43	0.56
Word2Vec-BiLSTM	55%	0.50	0.56	0.56	0.26	0.61	0.71	0.34	0.58	0.63
BERT-CNN	93%	0.92	0.93	0.95	0.92	0.92	0.95	0.92	0.92	0.95
BERT-RNN	93%	0.90	0.93	0.95	0.93	0.91	0.95	0.92	0.92	0.95
BERT-BiLSTM	93%	0.91	0.93	0.96	0.93	0.92	0.95	0.92	0.92	0.95

Table 9. Error evaluation.

Models	Mean Absolute Error	Mean Squared Error	Root Mean Square Error
CNN	0.1531	0.2305	0.4801
RNN	0.1253	0.1815	0.4260
BiLSTM	0.1242	0.1764	0.4200
Word2Vec-CNN	0.2822	0.2822	0.5313
Word2Vec-RNN	0.3456	0.3456	0.5879
Word2Vec-BiLSTM	0.2963	0.2963	0.5443
BERT-CNN	0.0796	0.1078	0.3283
BERT-RNN	0.0825	0.1133	0.3366
BERT-BiLSTM	0.0824	0.1128	0.3358

The confusion matrix of CNN indicates that 9261 out of 11,389 negative sentiments were correctly classified, while 12,673 out of 14,158 neutral sentiments were correctly classified and 15,710 out of 16,962 positive sentiments were correctly classified as shown in Figure 4 below.

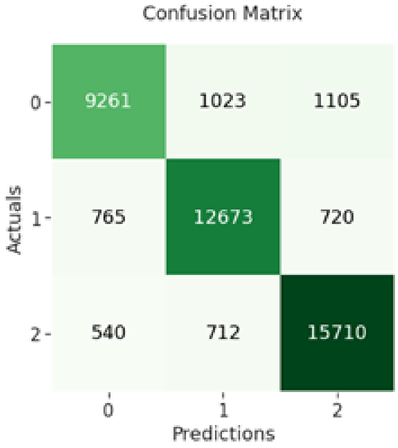

Figure 4. CNN confusion matrix.

The confusion matrix of CNN indicates that 9751 out of 11,389 negative sentiments were correctly classified, while 12,737 out of 14,158 neutral sentiments were correctly classified and 15,892 out of 16,962 positive sentiments were correctly classified as shown in Figure 5 below.

The confusion matrix of BiLSTM indicates that 9847 out of 11,221 negative sentiments were correctly classified, while 12,816 out of 14,272 neutral sentiments were correctly classified and 15,700 out of 17,040 positive sentiments were correctly classified as shown in Figure 6 below.

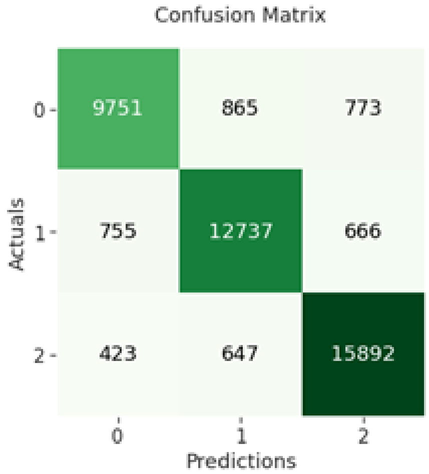

Figure 5. RNN confusion matrix.

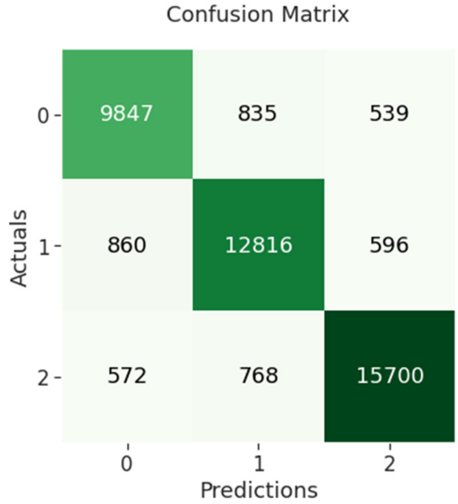

Figure 6. BiLSTM confusion matrix.

The confusion matrix of Word2Ve-BiLSTM indicates that 2901 out of 11,132 negative sentiments were correctly classified, while 8715 out of 14,393 neutral sentiments were correctly classified and 12,000 out of 16,984 positive sentiments were correctly classified as shown in Figure 7 below.

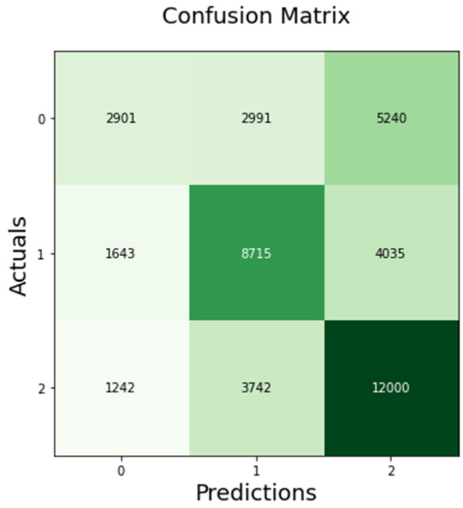

Figure 7. Word2Vec-BiLSTM confusion matrix.

The confusion matrix of BERT-CNN indicates that 10,348 out of 11,221 negative sentiments were correctly classified, while 13,145 out of 14,272 neutral sentiments were correctly classified and 16,254 out of 17,040 positive sentiments were correctly classified as shown in Figure 8 below.

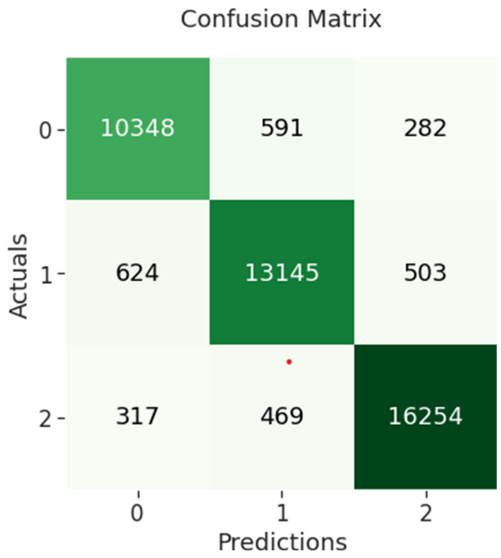

Figure 8. BERT-CNN confusion matrix.

The confusion matrix of BERT-RNN indicates that 10,478 out of 11,221 negative sentiments were correctly classified, while 13,003 out of 14,272 neutral sentiments were correctly classified and 16,199 out of 17,040 positive sentiments were correctly classified as shown in Figure 9 below.

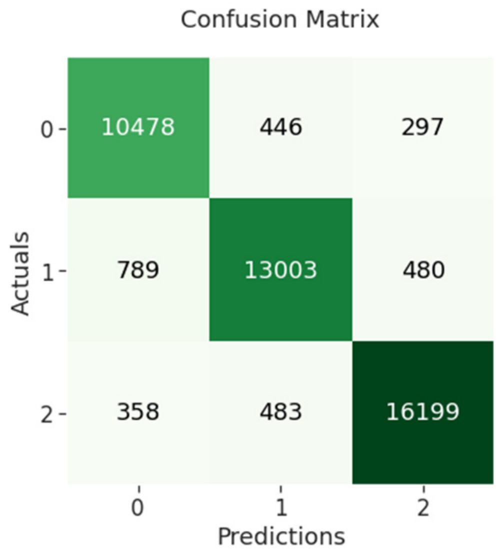

Figure 9. BERT-RNN confusion matrix.

The confusion matrix of BERT-BiLSTM indicates that 10,389 out of 11,221 negative sentiments were correctly classified, while 13,097 out of 14,272 neutral sentiments were correctly classified and 16,183 out of 17,040 positive sentiments were correctly classified as shown in Figure 10 below.

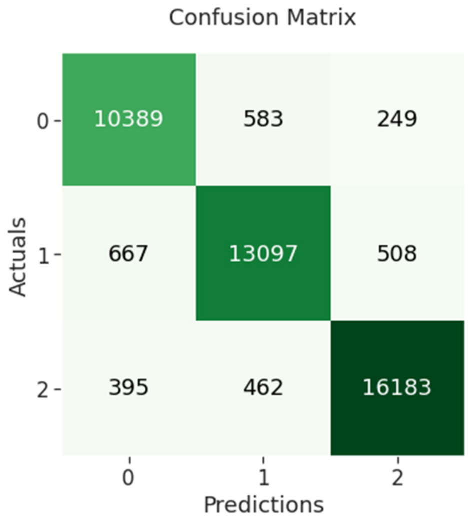

Figure 10. BERT-BiLSTM confusion matrix.

5. Conclusions

The traditional approach of natural language processing (NLP) with the use of Word2Vec, CNN, RNN, and BiLSTM has a few limitations of not capturing the deeper context of the word. The BERT has more understanding than traditional since the encoder process all inputs, which is the whole sentence, simultaneously so when building a context for a word, BERT will take into account the inputs before it and also the inputs after the word, while the Word2Vec restricts the ability to understand a word's meaning across two contexts and used in two different situations which in turn will be reflected on the output vector value for the word.

The combination of the transformer model BERT with CNN, RNN, and BiLSTM gives a state-of-the-art performance in terms of accuracy, recall, and precision.

Further works can be carried out to analyze sentiment on more data that is not sourced online because some information shared online can be shared by tourists or people with little or no understanding about the subject matter. Moreover, instead of sentiment, emotions such as happy, sad, and surprised can also be studied. Other transformer models such as RoBERTa can also be further investigated in further studies.

Author Contributions: Conceptualization, S.-C.N. and M.-F.L.; methodology, A.B.; software, A.B.; validation, M.-F.L., S.-C.N. and A.B.; formal analysis, M.-F.L.; investigation, S.-C.N.; resources, A.B.; data curation, A.B.; writing—original draft preparation, A.B.; writing—review and editing, S.-C.N. and M.-F.L.; visualization, A.B.; supervision, S.-C.N.; project administration, S.-C.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. These data can be found in the following URLs: https://www.kaggle.com/datasets/saurabhshahane/twitter-sentiment-dataset?select=Twitter_Data.csv (accessed on 30 September 2022). https://www.kaggle.com/datasets/seriousran/appletwittersentimenttexts?select=apple-twitter-sentiment-texts.csv (accessed on 30 September 2022). https://www.kaggle.com/datasets/surajkum1198/twitterdata?select=finalSentimentdata2.csv (accessed on 30 September 2022). https://www.kaggle.com/datasets/yasserh/twitter-tweets-sentiment-dataset?select=Tweets.csv (accessed on 30 September 2022). https://www.kaggle.com/code/toygarr/contextual-model-and-crawling-for-real-tweets/data?select=train.csv (accessed on 30 September 2022). https://www.kaggle.com/code/toygarr/contextual-model-and-crawling-for-real-tweets/data?select=test.csv (accessed on 30 September 2022).

Acknowledgments: The authors would also like to thank the anonymous reviewers for their insightful comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jayakody, J.P.U.S.D.; Kumara, B.T.G.S. Sentiment analysis on product reviews on twitter using Machine Learning Approaches. In Proceedings of the 2021 International Conference on Decision Aid Sciences and Application (DASA), Sakheer, Bahrain, 7–8 December 2021; pp. 1056–1061.
2. Pham, T.; Vo, D.; Li, F.; Baker, K.; Han, B.; Lindsay, L.; Pashna, M.; Rowley, R. Natural language processing for analysis of student online sentiment in a postgraduate program. *Pac. J. Technol. Enhanc. Learn.* **2020**, *2*, 15–30. [CrossRef]
3. Lamba, M.; Madhusudhan, M. Sentiment Analysis. In *Text Mining for Information Professionals*; Springer: Cham, Switzerland, 2021; pp. 191–211.
4. Suhasini, M.; Srinivasu, B. Emotion detection framework for twitter data using supervised classifiers. In *Data Engineering and Communication Technology*; Springer: Singapore, 2020; pp. 565–576.
5. Kepios. Available online: https://kepios.com/ (accessed on 15 December 2022).
6. Comito, C.; Falcone, D.; Talia, D. A Peak Detection Method to Uncover Events from Social Media. In Proceedings of the 2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA), Tokyo, Japan, 19–21 October 2017; pp. 459–467. [CrossRef]
7. Comito, C.; Pizzuti, C.; Procopio, N. Online Clustering for Topic Detection in Social Data Streams. In Proceedings of the 2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI), San Jose, CA, USA, 6–8 November 2016; pp. 362–369. [CrossRef]
8. Khurana, D.; Koli, A.; Khatter, K.; Singh, S. Natural language processing: State of the art, current trends and challenges. *Multimed. Tools Appl.* 2022, *in press*. [CrossRef] [PubMed]
9. Jain, A.; Kulkarni, G.; Shah, V. Natural Language Processing. *Int. J. Comput. Sci. Eng.* **2018**, *6*, 161–167. [CrossRef]
10. Dai, C.; Che, H.; Leung, M.F. A neurodynamic optimization approach for L1 minimization with application to compressed image reconstruction. *Int. J. Artif. Intell. Tools* **2021**, *30*, 2140007. [CrossRef]
11. Leung, M.F.; Wang, J.; Che, H. Cardinality-constrained portfolio selection via two-timescale duplex neurodynamic optimization. *Neural Netw.* **2022**, *153*, 399–410. [CrossRef] [PubMed]
12. Leung, M.F.; Wang, J.; Li, D. Decentralized robust portfolio optimization based on cooperative-competitive multiagent systems. *IEEE Trans. Cybern.* **2022**, *52*, 12785–12794. [CrossRef] [PubMed]
13. Chen, K.; Che, H.; Li, X.; Leung, M.F. Graph non-negative matrix factorization with alternative smoothed L0 regularizations. *Neural Comput. Appl.* **2022**, *in press*. [CrossRef]
14. Che, H.; Wang, J.; Cichocki, A. Bicriteria sparse nonnegative matrix factorization via two-timescale duplex neurodynamic optimization. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *in press*. [CrossRef] [PubMed]
15. Wang, J.; Wang, J.; Han, Q.L. Neurodynamics-based model predictive control of continuous-time under-actuated mechatronic systems. *IEEE/ASME Trans. Mechatron.* **2021**, *26*, 311–322. [CrossRef]
16. Hinton, G.E.; Osindero, S.; Teh, Y.W. A fast learning algorithm for deep belief nets. *Neural Comput.* **2006**, *18*, 1527–1554. [CrossRef] [PubMed]
17. Vateekul, P.; Koomsubha, T. A study of sentiment analysis using deep learning techniques on Thai Twitter data. In Proceedings of the 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE), Khon Kaen, Thailand, 13–15 July 2016.
18. Cao, W.; Qian, S.; Wu, S.; Wong, H.S. Unsupervised multi-task learning with hierarchical data structure. *Pattern Recognit.* **2019**, *86*, 248–264. [CrossRef]
19. Cao, W.; Zhang, Z.; Liu, C.; Li, R.; Jiao, Q.; Yu, Z.; Wong, H.S. Unsupervised discriminative feature learning via finding a clustering-friendly embedding space. *Pattern Recognit.* **2022**, *129*, 108768. [CrossRef]
20. Duan, Y.; Chen, N.; Bashir, A.K.; Alshehri, M.D.; Liu, L.; Zhang, P.; Yu, K. A Web Knowledge-Driven Multimodal Retrieval Method in Computational Social Systems: Unsupervised and Robust Graph Convolutional Hashing. *IEEE Trans. Comput. Soc. Syst.* **2022**, *in press*. [CrossRef]
21. Zhang, L.; Wang, S.; Liu, B. Deep learning for sentiment analysis: A survey. *WIREs Data Min. Knowl. Discov.* **2018**, *8*, e1253. [CrossRef]
22. Chakravarthi, B.; Ng, S.C.; Ezilarasan, M.R.; Leung, M.F. EEG-based emotion recognition using hybrid CNN and LSTM classification. *Front. Comput. Neurosci.* **2022**, *16*, 1019776. [CrossRef] [PubMed]
23. Halbert, J.; Kaser, L. Deep learning: Inquiring communities of practice. *Educ. Can.* **2006**, *46*, 43–45.
24. Bhagat, C.; Mane, D. Text categorization using sentiment analysis. In Proceedings of the International Conference on Computational Science and Applications, Saint Petersburg, Russia, 1–4 July 2019; Springer: Singapore, 2020; pp. 361–368.
25. Chiorrini, A.; Diamantini, C.; Mircoli, A.; Potena, D. Emotion and sentiment analysis of tweets using BERT. In Proceedings of the EDBT/ICDT Workshops, Nicosia, Cyprus, 23–26 March 2021.

26. Huang, H.; Jing, X.Y.; Wu, F.; Yao, Y.F.; Zhang, X.Y.; Dong, X.W. DCNN-Bigru text classification model based on BERT embedding. In Proceedings of the 2019 IEEE International Conferences on Ubiquitous Computing & Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS), Shenyang, China, 21–23 October 2019; pp. 632–637.
27. Ouyang, X.; Zhou, P.; Li, C.; Liu, L. Sentiment Analysis Using Convolutional Neural Network. In Proceedings of the 2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, Liverpool, UK, 26–28 October 2015; pp. 2359–2364.
28. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.
29. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.; Kaiser, L.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–11.
30. Zorn, J. Deep Learning for NLP: Word Embeddings. [Online] Medium. 2020. Available online: https://towardsdatascience.com/deep-learning-for-nlp-word-embeddings-4f5c90bcdab5 (accessed on 15 December 2022).
31. Basha, S.S.; Dubey, S.R.; Pulabaigari, V.; Mukherjee, S. Impact of fully connected layers on performance of convolutional neural networks for image classification. *Neurocomputing* **2020**, *378*, 112–119. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.